# 基于匈牙利匹配算法的钓鱼网页检测方法

张卫丰[1] 周毓明[2] 许 蕾[2] 徐宝文[2]

[1](南京邮电大学计算机学院 南京 210003)

[2](南京大学计算机科学与工程系 南京 210093)

**摘 要** 如何快速有效地计算网页的相似性是发现钓鱼网页的关键.现有的钓鱼网页检测方法在检测效果上依然存在较大的提升空间.文中提出基于匈牙利匹配的钓鱼网页检测模型,该模型首先提取渲染后网页的文本特征签名、图像特征签名以及网页整体特征签名,比较全面地刻画了网页访问后的特征;然后通过匈牙利算法计算二分图的最佳匹配来寻找不同网页签名之间匹配的特征对,在此基础上能够更加客观地度量网页之间的相似性,从而提高钓鱼网页的检测效果.一系列的仿真实验表明文中方法可行,并具有较高的准确率和召回率.

**关键词** 钓鱼网页;网页特征;匈牙利匹配算法;相似性;网页签名

**中图法分类号** TP391 **DOI 号**: 10.3724/SP.J.1016.2010.01963

# A Method of Detecting Phishing Web Pages Based on Hungarian Matching Algorithm

ZHANG Wei-Feng[1] ZHOU Yu-Ming[2] XU Lei[2] XU Bao-Wen[2]

[1](*Scholl of Computer, Nanjing University of Posts and Telecommunications, Nanjing* 210003)

[2](*Department of Computer Science and Engineering, Nanjing University, Nanjing* 210093)

**Abstract** It is the key problem for detecting the phishing pages how to quickly and efficiently to calculate the similarity of web pages. There is still a large space to improve the detecting efficiency in current anti phishing method. A method of detecting phishing web pages based on bipartite graph matching is brought forward. In this model, the signature of text, the signature of images, and the signature of the overall web page are extracted. Then, by the Hungarian algorithm, the best match in the bipartite graph(signatures in different pages) is found. The pairs of features are then used to measure the similarity between pages in an more objective way, thereby the effectiveness of phishing page detection is improved. A series of simulation experiments show that this method is feasible with high precision and recall rate.

**Keywords** antiphishing; web metric; bipartite graph matching; similarity; web page signature

## 1 引 言

互联网应用不断普及,越来越多的人开始使用一些在线的互联网服务,如网上银行、在线购物等.在线互联网服务给广大用户带来很大便利性的同时,以骗取用户账号为目的钓鱼网站迅速增多.钓鱼网站一般通过模仿真实的网站界面来欺骗用户:当

收稿日期:2010-08-22.本课题得到国家自然科学基金(60703086,60873050,60803008,60973046、苏州大学江苏省计算机信息处理技术重点实验室基金(KJS0714)及江苏省高校自然科学研究计划(09KJB520012)资助.**张卫丰**,男,1975 年生,博士,副教授,主要研究方向为 Web 信息获取、Web 数据挖掘、Spam 检测技术.E-mail: wfzhang@yahoo.com. **周毓明**,男,1974 年生,博士,教授,主要研究领域为软件度量、软件测试. **许 蕾**,女,1974 年生,博士,副教授,主要研究方向为 Web 测试、Web Service 测试、Web 技术. **徐宝文**,男,1961 年生,教授,博士生导师,主要研究领域为软件测试、软件度量、Web 技术.

用户在钓鱼网站输入敏感信息,比如用户名口令、银行卡账号及密码等信息后,钓鱼网站将窃取并非法利用用户输入的这些敏感信息,从而给用户带来巨大侵害.钓鱼网站往往通过垃圾邮件的形式随机发送给用户信息,这些邮件往往欺骗那些没有网络安全经验的人,使得他们相信邮件来自合法的组织,一般这些邮件都以某种原因要求用户更新信息、输入用户名密码等.虽然大多数用户都具有一定的安全经验,而且很多邮件网关可以过滤掉大部分的垃圾邮件,但是总有一些用户成为钓鱼网站的牺牲品.根据网站 phishtank 的统计,钓鱼网站近年来快速发展,到 2008 年 10 月的两年中,该网站登记了超过 100 万个钓鱼网站,而且每天都有大量新的钓鱼网站被发现,钓鱼网站欺骗行为正有愈演愈烈的趋势.目前防止钓鱼网站的方法主要有基于邮件的方法、基于黑名单的方法、基于网站特征的方法和基于信息流的方法[1-4].其中,基于邮件的方法主要利用反垃圾邮件技术过滤掉包含钓鱼网站的链接.基于黑名单的方法通过收集的钓鱼网站 URL 构建黑名单,在访问当前 URL 时将与黑名单中的 URL 比较,如果出现在黑名单中,将阻止用户对当前 URL 访问,例如微软的 Internet Explorer(IE)7 和 Google 的浏览器[5]都内建了基于黑名单的反钓鱼网站技术,另外一些软件如 NetCraft[6]、SiteAdvisor[7]等通过加载成浏览器工具条的形式来根据黑名单过滤钓鱼网站,但是黑名单方法需要及时保存该名单的更新才能对付最新的钓鱼网站.

为了保证互联网服务的安全,迫切需要快速有效的钓鱼网站检测技术[8].Liu 等提出利用 DOM 树之间的相似性来检测钓鱼网页[9-10],该方法认为钓鱼网页为了欺骗用户,往往展现比较相近的界面,否则不易欺骗用户,所以真实网页与钓鱼网页在布局上应该非常接近.Liu 等利用钓鱼网站与真实网站之间的视觉相似性来检测钓鱼网站[2,9-12],该类方法首先提取真实网页与待检测网页的图像特征,然后计算相似性,当相似性大于某一设定阈值时,则认为待检测网页为钓鱼网页;Rosiello 等通过比较网页 HTML 标记之间的相似性计算网页间相似性[13].以上方法尽管可以在一定程度上检测出钓鱼网站,但是这些方法大多将单个网页请求作为比较对象,另外在计算网页相似度过程中没有选择合适的特征对进行比较,导致网页相似度的计算精度低,直接影响钓鱼网页检测的精度和召回率.针对以上问题,本文提出基于匈牙利匹配算法的钓鱼网站检测技术,首先从渲染后的网页中提取网页的文本特征和图像特征,从而能够整体上刻画浏览器中所显示网页的特征;其次在网页签名特征比较过程中,通过匈牙利算法求得适合比较的特征对,从而提高了相似度比较的效果.

本文将首先介绍网页渲染特征模型,包括网页文本特征、网页图片特征以及网页全局图片特征;然后提出基于匈牙利匹配的网页签名特征匹配方法,在获得匹配特征对的基础上计算网页相似度;最后通过仿真实验获得模型中的参数,进而比较采用匈牙利最佳匹配计算相似度的方法与其它方法.

## 2 网页渲染特征模型

浏览器在访问网页过程中,根据网页内部的链接以及脚本语言来决定是否发出其它 URL 请求,在该网页所包含的资源返回后,浏览器将把这些资源(文字、图片等)按照各自的属性渲染在浏览器中,最终形成图文并茂的网页.钓鱼网页是为了让人们在通过浏览器访问时相信其就是那些被仿冒的网页,其仿冒主要体现在渲染特征的相似性.因此钓鱼网站的基本检测过程分为如下 3 步:

1. 获取钓鱼网页及其所需要的其它相关资源,记为 $url$;

2. 获取钓鱼网页渲染后的特征(也可称为签名),记为 $S(url)$;

3. 将 $S(url)$与保存的对应钓鱼目标网页(钓鱼网页所要模仿的网页)的签名 $S(urlT)$进行比较,如果两者的相似度大于某一设定的阈值,则认为网页 $url$ 是以网页 $urlT$ 为目标的钓鱼网页,发出报警.

从以上钓鱼网页检测步骤可知,要提高钓鱼网页的检测效果,一是要抽取渲染后网页的特征,二是要提高特征相似度计算的精度.渲染后网页的特征有文本特征、网页内图片特征和网页整体视觉特征等.为了便于下文对特征相似度计算,先形式化定义网页特征.

浏览器在下载网页及相关图片后,把网页解析成 HTML DOM 树.网页中的可见文本保存在 DOM 树的叶节点上.

**定义 1.** 网页 $url$ 的文本特征,由多个六元组向量组成,记为$\langle t_1, t_2, \cdots, t_{n(t,url)} \rangle$, $t_i = \langle t_{i1}, t_{i2}, t_{i3}, t_{i4}, t_{i5}, t_{i6} \rangle$,其中 $t_{i1}$ 为对应文本内容;$t_{i2}$ 为文本的前景色;$t_{i3}$ 为文本的背景色;$t_{i4}$ 为文本的字体大小;$t_{i5}$ 为文本的字体名称;$t_{i6}$ 表示文本在网页中的位置;$n(t,url)$表示网页 $url$ 中文本特征的数量.

定义 1 中抽取了网页中的文本可见特征,网页

除了包含文本，图片也是最常见的一种元素，定义网页中每个图片的特征如下.

**定义 2.** 网页 $url$ 中的图片特征，由多个五元组向量组成，记为$\langle m_1, m_2, \cdots, m_{n(m,url)}\rangle$，$m_i=\langle m_{i1}, m_{i2}, m_{i3}, m_{i4}, m_{i5}\rangle$，其中 $m_{i1}$ 为图片的 $src$ 属性；$m_{i2}$ 为图片的面积；$m_{i3}$ 为图片的颜色直方图；$m_{i4}$ 为文本在网页中的位置；$m_{i5}$ 为小波特征.

文字特征和图片特征从局部反应了网页的特征，这些文字和图片由标记语言标记，浏览器通过这些标记语言来布局文字和图片，最终呈现给用户渲染后的网页. 渲染后的网页可以看成全局图像，这样可以从全局图像的角度来提取渲染后网页的特征.

**定义 3.** 网页 $url$ 的全局图像(Overall)特征，表示为$\langle\langle C_1, Cent_1, N_{C_1}\rangle, \langle C_2, Cent_2, N_{C_2}\rangle, \cdots, \langle C_{ND}, Cent_{ND}, N_{C_{ND}}\rangle\rangle$，其中 $C_1, C_2, \cdots, C_{ND}$ 表示不同的颜色；$Cent_i=\frac{1}{N_{C_i}}\sum_{k=1}^{N_{C_i}} coord(k,i)$，$coord(k,i)$ 表示具有颜色 $C_i$ 的第 $k$ 个像素的坐标，$Cent_i$ 即表示具有颜色 $C_i$ 的像素点的中心点坐标；$N_{C_i}$ 表示具有颜色 $C_i$ 的像素点个数.

定义 3 中的 Overall 特征根据每种颜色的数量来选择一些颜色作为 Overall 的签名特征，即不同的图片所选择的颜色可能不同，签名特征的长度也可能不同.

根据定义 1～3，可以进一步定义网页签名.

**定义 4.** 网页签名 $S(url)=\langle\langle t_1, t_2, \cdots, t_{n(t,url)}\rangle, \langle m_1, m_2, \cdots, m_{n(m,url)}\rangle, \langle\langle C_{s1}, Cent_{s1}, N_{C_{s1}}\rangle, \langle C_{s2}, Cent_{s2}, N_{C_{s2}}\rangle, \cdots, \langle C_{sn}, Cent_{sn}, N_{C_{sn}}\rangle\rangle\rangle$，其中 $N_{C_{s1}}\geq N_{C_{s2}}\cdots\geq N_{C_{sn}}$，网页 $url$ 的 Overall 特征选取出现频率最高的前 $sn$ 种颜色.

在提取网页渲染特征的基础上，可以计算网页签名相似性，网页签名相似性由文本签名特征相似性、图片签名特征相似性以及全局图像签名特征相似性组成. 提高这些相似性计算的精度将直接影响钓鱼网页的检测效果.

## 3 基于匈牙利匹配算法的 Web 网页相似性度量

从定义 4 可知，对于不同的网页可以抽取对应的网页签名，而不同网页的签名在向量长度、特征等方面不尽相同. 对于不等长特征的相似度计算有简单比较法(All match，简记为 Simple)[13]、贪婪算法(N largest match，简记为 Normal)[14] 和 EMD 方法[15-16]等. 简单比较法通过网页标记来确定哪些特征进行比较；贪婪算法通过选择相似矩阵中具有前几个最大值的对应特征作为比较对象；而 EMD 方法将不同签名之间的相似性比较问题转化为“运输问题”. 尽管这些方法可以近似计算签名之间的相似性，但是在计算过程中没有找匹配的特征对进行比较，这将直接影响网页签名相似度计算的精度. 本文将网页签名之间的相似性计算建立在匹配的特征对基础上，即首先将不同签名的特征进行匹配，得到匹配的特征对，然后通过匹配的特征对之间的相似性计算网页签名的相似性. 本文将网页签名特征的匹配问题建模成求二分图的最佳匹配问题，二分图的最佳匹配问题可以利用匈牙利算法(KM match，简记为 KM)计算. 下文给出该模型，并证明其可行性.

**定义 5.** 网页签名匹配问题. 给定两个网页签名 $X=\langle x_1, x_2, \cdots, x_m\rangle$ 和 $Y=\langle y_1, y_2, \cdots, y_n\rangle$，$Sim(x,y)$ 表示特征 $x$ 与特征 $y$ 的相似性($x\in X$，$y\in Y$)，则网页签名匹配问题可以定义为：求一个 $X$ 与 $Y$ 的子集 $X'$ 与 $Y'$，使得存在 $X'$ 到 $Y'$ 的双射 $f: X'\leftrightarrow Y'$，满足$\sum_{x\in X} Sim(x, f(x))$取得最大值.

为了求解网页签名匹配问题，我们借鉴二分图的概念，将网页签名匹配问题建模成一个二分图 $G=(X,Y,E)$，$X$ 和 $Y$ 分别对应两个网页的签名；边集 $E$ 按照如下规则构造：如果 $Sim(x,y)>0$，$x\in X$，$y\in Y$，则在二分图 $G$ 中对应的两个顶点 $x$ 与 $y$ 之间连一条边 $(x,y)$，并设置该边的权重 $w_{xy}=Sim(x,y)$. 通过该二分图模型，网页签名的匹配问题就转化为在求二分图 $G$ 上的从顶点 $X$ 到 $Y$ 的匹配问题.

给定一个二分图 $G$，在 $G$ 的一个子图 $M$ 中，$M$ 的边集中的任意两条边都不依附于同一个顶点，则称 $M$ 是一个匹配. 选择这样的边数最大的子集称为图的最大匹配问题. 二分图的最大匹配问题可以通过 KM(Kuhn-Munkres)算法来求解，该算法是一个经典求解二分图的最佳匹配的算法[17-18]. 不过二分图的最佳匹配针对与顶点数相等的完全二部图来计算，而网页签名的长度不同，无法直接使用二分图的最佳匹配算法，为此需要在 $|X|$ 与 $|Y|$ 不相等时对二部图进行扩展.

**定义 6.** 扩展二分完全图 $G'=(X',Y',E')$，其中 $G'$ 在 $G$ 的基础上通过如下方法扩展而成：在顶点数少的一侧添加虚顶点，使得 $X'$ 中顶点数和 $Y'$ 中顶点数相等；$E'$ 在 $E$ 的基础上扩展，如果 $X'$ 中顶点 $X'_i$ 和 $Y'$ 中顶点 $Y'_j$ 之间没有边，则添加一条权值

为 0 的虚边$(i,j)$.

定义 6 中扩展二分完全图$G'$的最佳匹配问题可以直接使用匈牙利算法得到最佳匹配,该最佳匹配经过适当裁剪后是否可以求得二分图 $G$ 的最佳匹配,该方法由定理 1 得到保证.

**定理 1.** 扩展二分完全图$G'=(X',Y',E')$的最佳匹配 $M'$中删除虚顶点及虚边后得到的匹配 $M$ 为覆盖 $X$ 与 $Y$ 中顶点数较少顶点集的 $G$ 的权和最大匹配.

证明. 对于$|X|=|Y|$的情况,由于加入的虚边的权值为 0,即便删除了虚边,对 $M$ 的权重没有影响,根据完全二分图的最佳匹配的定义和定理,在该情况下定理 1 成立;这里针对$|X|\neq|Y|$的情况,不妨设$|X|<|Y|$. 记 $G'$中虚顶点的集合为 $X^0$,虚边的集合为 $E^0$. 则需要证明:(1) $M$ 覆盖 $X$;(2) $M$ 的权和最大.

对于(1),根据 $M$ 的生成规则,由于 $X=X'-X^0$,所以成立;

对于(2),假设存在权和更大的 $G$ 的匹配 $M''$,则 $M''$覆盖 $X$ 中的顶点,设 $Y$ 中未被覆盖的顶点集为 $Y^0$,则有$|X^0|=|Y^0|$;可以在 $X^0$ 与 $Y^0$ 之间一一对应添加$|Y^0|$条权重为 0 的边 $E^1$,则 $M''\cup E^1$成为 $G'$的匹配,而根据 $M$ 的构造过程知,$M$ 的权和与 $M'$的权和相等,即得到 $M''\cup E^1$的权和大于 $M'$的权和,这与 $M'$为 $G'$的最佳匹配相矛盾,所以假设不成立,得证. 证毕.

根据定理 1,可以通过先计算扩展二分完全图来获得 $G$ 的权和最大匹配,在此基础上可以计算签名 $X$ 和 $Y$ 之间的相似性(如算法 1). 算法 1 中第 1 步通过比较 $X$ 和 $Y$ 的顶点数量是否相等,如相等,则进入步 2,直接使用 KM 算法求解 $G$ 的最佳匹配 $M$;如不相等,则进入步 3～5,步 3 将 $G$ 根据定义 6 添加虚顶点及虚拟边,生成扩展二分完全图 $G'$,步 4 对 $G'$利用 KM 算法求解得到其最佳匹配 $M'$,步 5 根据定理 1 求得 $G$ 的权和最大匹配.

**算法 1.** 签名相似度计算.

输入:$G=(X,Y,E)$

输出:$X$ 与 $Y$ 的相似度

1. IF ($|X|==|Y|$){
2. 使用 KM 算法获得 $G$ 的最佳匹配 $M$;
   }
   ELSE{
3. 根据定义 6 生成 $G$ 的扩展二分完全图$G'$;
4. 使用 KM 算法获得 $G'$的最佳匹配 $M'$;
5. 在 $M'$的基础上根据定理 1 获得 $G$ 的权和最大匹配 $M$;
   }
6. 根据 $M$ 中的边的权重相加取平均即为 $X$ 与 $Y$ 的相似度.

为了计算二分图 $G$ 的最大权和匹配,需要知道顶点间每条边的权重. 对于网页签名来讲,需要知道钓鱼网页与目标网页之间文本特征相似性、图片特征相似性和 Overall 特征相似性,下文将分别给出计算方法.

### 3.1 文本特征相似性计算

对于网页签名 $S(url_1)$中的文本特征 $t_i$ 与 $S(url_2)$中的文本特征 $t_j$的相似性计算如下:

文本串 $T$ 与 $\overline{T}$ 之间的相似性:

$$S_T(T,\overline{T})=1-\frac{D_{LT}(T,\overline{T})}{\max(Len(T),Len(\overline{T}))},$$

其中,$D_{LT}(T,\overline{T})$为 $T$ 与 $\overline{T}$ 之间的 $L1$ 距离;$Len(T)$与 $Len(\overline{T})$分别表示文本串 $T$ 与 $\overline{T}$ 的长度.

颜色 $C$ 与 $\overline{C}$ 之间的相似性:

$$S_C(C,\overline{C})=1-\frac{D_{LC}(C,\overline{C})}{ColorNUM},$$

其中,$D_{LC}(C,\overline{C})=|R-\overline{R}|+|G-\overline{G}|+|B-\overline{B}|$,$ColorNUM$ 表示总共的颜色数,对于 $3\times8$ 位 RGB 的像素点来说,$ColorNUM=3\times256$.

字体大小 $F$ 与 $\overline{F}$ 之间的相似度:

$$S_{FS}(F,\overline{F})=1-\frac{|F-\overline{F}|}{\max(F,\overline{F})}.$$

字体 $F_F$与 $\overline{F}_F$之间的相似度:

$$S_{FF}(F_F,\overline{F}_F)=\begin{cases}1, & F_F=\overline{F}_F\\ 0, & F_F\neq\overline{F}_F\end{cases}.$$

位置 $P$ 与 $\overline{P}$ 之间的相似度(见算法 1):

$$S_P(P,\overline{P})=1-\frac{D_P(P,\overline{P})}{M_d},$$

其中,$D_P(P,\overline{P})$为像素 $P$ 与 $\overline{P}$ 之间的欧氏距离,$M_d$为浏览器所显示页面的对角线长度.

对文本串相似性、颜色相似性、字体大小相似性、字体相似性以及位置相似性进行加权平均,得到 $S(url_1)$中的文本特征 $t_i$与 $S(url_2)$中的文本特征 $t_j$的相似性.

$$\begin{aligned}S^{\mathrm{T}}_{i,j}=&\alpha^{\mathrm{T}}_T\times S_T(T,\overline{T})+\alpha^{\mathrm{T}}_{CF}\times S_C(C_F,\overline{C}_F)+\\&\alpha^{\mathrm{T}}_{CB}\times S_C(C_B,\overline{C}_B)+\alpha^{\mathrm{T}}_{FS}\times S_{FS}(F,\overline{F})+\\&\alpha^{\mathrm{T}}_{FF}\times S_{FF}(F_F,\overline{F}_F)+\alpha^{\mathrm{T}}_P\times S_P(P,\overline{P}),\end{aligned}$$

其中,$\alpha^{\mathrm{T}}_T$,$\alpha^{\mathrm{T}}_{CF}$,$\alpha^{\mathrm{T}}_{CB}$,$\alpha^{\mathrm{T}}_{FS}$,$\alpha^{\mathrm{T}}_{FF}$和 $\alpha^{\mathrm{T}}_P$分别表示在计算文本特征相似性公式中,文本串相似性、前景色相似性、背景色相似性、字体大小相似性、字体相似性和文字位置相似性的权重,满足 $\alpha^{\mathrm{T}}_T+\alpha^{\mathrm{T}}_{CF}+\alpha^{\mathrm{T}}_{CB}+\alpha^{\mathrm{T}}_{FS}+$

$\alpha_{FF}^{\mathrm{T}}+\alpha_{P}^{\mathrm{T}}=1$；$C_F$，$\overline{C}_F$分别表示两个文本的前景色；$C_B$，$\overline{C}_B$分别表示两文本的背景色.

### 3.2　图片特征相似性计算

对于网页签名 $S(url_1)$中的图片特征 $m_i$ 与 $S(url_2)$中的图片特征 $m_j$的相似性计算如下：

图片 *src* 属性 $F_{src}$与 $\overline{F}_{src}$的相似性：

$$S_{src}(F_{src},\overline{F}_{src})=\begin{cases}1, & F_{src}=\overline{F}_{src}\\ 0, & F_{src}\neq\overline{F}_{src}\end{cases}.$$

图片面积 $A$ 与 $\overline{A}$ 的相似性：

$$S_A(A,\overline{A})=1-\frac{|A-\overline{A}|}{\max(A,\overline{A})}.$$

图片颜色直方图向量 $\boldsymbol{C}_c$与 $\overline{\boldsymbol{C}}_c$的相似性：

$$S_{CC}(\boldsymbol{C}_c,\overline{\boldsymbol{C}}_c)=\frac{\boldsymbol{C}_c\cdot\overline{\boldsymbol{C}}_c}{|\boldsymbol{C}_c|\times|\overline{\boldsymbol{C}}_c|}.$$

图片小波特征 $\boldsymbol{W}$ 与 $\overline{\boldsymbol{W}}$ 的相似性：

$$S_W(\boldsymbol{W},\overline{\boldsymbol{W}})=\frac{\boldsymbol{W}\cdot\overline{\boldsymbol{W}}}{|\boldsymbol{W}|\times|\overline{\boldsymbol{W}}|}.$$

对两个图片的 *src* 特征相似性、面积特征相似性、颜色直方图相似性以及位置相似性加权平均，得到 $S(url_1)$中图片特征 $m_i$与 $S(url_2)$中图片特征 $m_j$的相似性.

$$S_{i,j}^{\mathrm{I}}=\alpha_{src}^{\mathrm{I}}\times S_{src}(F_{src},\overline{F}_{src})+\alpha_A^{\mathrm{I}}\times S_A(A,\overline{A})+\alpha_{CC}^{\mathrm{I}}\times S_{CC}(\boldsymbol{C}_c,\overline{\boldsymbol{C}}_c)+\alpha_W^{\mathrm{I}}S_W(\boldsymbol{W},\overline{\boldsymbol{W}})+\alpha_P^{\mathrm{I}}S_P(P,\overline{P}),$$

其中 $\alpha_{src}^{\mathrm{I}}$，$\alpha_A^{\mathrm{I}}$，$\alpha_{CC}^{\mathrm{I}}$，$\alpha_W^{\mathrm{I}}$和 $\alpha_P^{\mathrm{I}}$分别表示在计算图片相似性过程中，图片 *src* 属性相似性、图片面积相似性、图片颜色直方图向量相似性、小波特征向量相似性和图片位置相似性的权重，满足 $\alpha_{src}^{\mathrm{I}}+\alpha_A^{\mathrm{I}}+\alpha_{CC}^{\mathrm{I}}+\alpha_W^{\mathrm{I}}+\alpha_P^{\mathrm{I}}=1$.

### 3.3　网页全局图像特征相似性计算

对于网页签名 $S(url_1)$中的 Overall 特征$\langle C_i,Cent_i,N_{C_i}\rangle$与 $S(url_2)$中的 Overall 特征$\langle C_j,Cent_j,N_{C_j}\rangle$的相似性计算如下：

颜色相似性：

$$S_{\mathrm{color}}(C_i,C_j)=\frac{\sqrt{(C_i-C_j)\times(C_i-C_j)^{\mathrm{T}}}}{Max_{\mathrm{color}}},$$

其中 $Max_{\mathrm{color}}$表示最大的颜色数.

位置相似性：

$$S_{\mathrm{center}}(Cent_i,Cent_j)=\frac{\sqrt{(Cent_i-Cent_j)\times(Cent_i-Cent_j)^{\mathrm{T}}}}{\sqrt{w^2+h^2}},$$

其中 $w$ 和 $h$ 分别为标准后图像的宽度和高度.

对应颜色数量相似性：

$$S_{\mathrm{number}}(N_{C_i},N_{C_j})=\frac{|N_{C_i}-N_{C_j}|}{\max(N_{C_i},N_{C_j})}.$$

$\langle C_i,Cent_i,N_{C_i}\rangle$与$\langle C_j,Cent_j,N_{C_j}\rangle$的相似性：

$$S^{\mathrm{O}}(\langle C_i,Cent_i,N_{C_i}\rangle,\langle C_j,Cent_j,N_{C_j}\rangle)=\alpha_{\mathrm{color}}^{\mathrm{O}}\times S_{\mathrm{color}}(C_i,C_j)+\alpha_{\mathrm{conter}}^{\mathrm{O}}\times S_{\mathrm{center}}(Cent_i,Cent_j)+\alpha_{\mathrm{number}}^{\mathrm{O}}\times S_{\mathrm{number}}(N_{C_i},N_{C_j}),$$

其中，$\alpha_{\mathrm{color}}^{\mathrm{O}}$，$\alpha_{\mathrm{conter}}^{\mathrm{O}}$和 $\alpha_{\mathrm{number}}^{\mathrm{O}}$分别表示在计算 Overall 的特征相似性中，颜色相似性、位置相似性和对应颜色数量相似性的权重，满足 $\alpha_{\mathrm{color}}^{\mathrm{O}}+\alpha_{\mathrm{conter}}^{\mathrm{O}}+\alpha_{\mathrm{number}}^{\mathrm{O}}=1$.

通过以上方法，可以计算网页签名 $S(url_1)$与 $S(url_2)$中文本特征相似矩阵 $\boldsymbol{S}^{\mathrm{T}}$、图片特征相似矩阵 $\boldsymbol{S}^{\mathrm{I}}$以及全局图片特征相似矩阵 $\boldsymbol{S}^{\mathrm{O}}$，根据这些相似矩阵最终计算网页签名之间的相似性. 从相似矩阵抽取相似性有多种方法：最简单的方法是把矩阵元素取平均值(平均值法 All match)，这种方法计算简单，但是精度较低；取矩阵中最大的 $n$ 个相似对，对这 $n$ 个相似对对应的相似度取平均值(最大 $n$ 元素法 $N$ largest match)，该方法相对简单，但是考虑的相似对不尽合理，导致误差较大. 本文利用二分图模型，在相似矩阵的基础上通过求二分图的最佳匹配来获得匹配特征对，在此基础上抽取相似矩阵中对应匹配特征对之间的相似性(KM match).

最终网页签名 $S(url_1)$与 $S(url_2)$的相似性：

$$s=\alpha^{\mathrm{T}}\times s^{\mathrm{T}}+\alpha^{\mathrm{I}}\times s^{\mathrm{I}}+\alpha^{\mathrm{O}}\times s^{\mathrm{O}},$$

其中 $\alpha^{\mathrm{T}}+\alpha^{\mathrm{I}}+\alpha^{\mathrm{O}}=1$，$s^{\mathrm{T}}$，$s^{\mathrm{I}}$，$s^{\mathrm{O}}$分别表示网页签名中文本相似度、图片相似度和 Overall 相似度.

从以上方法知，要计算最终的网页签名相似性需要设定如下参数：$\alpha_T^{\mathrm{T}}$，$\alpha_{CF}^{\mathrm{T}}$，$\alpha_{CB}^{\mathrm{T}}$，$\alpha_{FS}^{\mathrm{T}}$，$\alpha_{FF}^{\mathrm{T}}$，$\alpha_P^{\mathrm{T}}$，$\alpha_{src}^{\mathrm{I}}$，$\alpha_A^{\mathrm{I}}$，$\alpha_{CC}^{\mathrm{I}}$，$\alpha_P^{\mathrm{I}}$，$\alpha_{src}^{\mathrm{I}}$，$\alpha_A^{\mathrm{I}}$，$\alpha_{CC}^{\mathrm{I}}$，$\alpha_P^{\mathrm{I}}$. 本文将通过实验来获得这些参数.

若待检测网页与目标网页的网页签名的相似性大于设定的阈值，则认为该待检测网页为钓鱼网页. 检测流程如图 1 所示，浏览器插件首先记录需要输入用户名和密码的网页 URL、用户名和密码，同时提取该网页的签名特征保存在特征库中；当用户访问当前网页时用户输入同样的用户名和密码，但是 URL 不一致的时候，则启动钓鱼网页检测插件，该检测插件提取可能被仿冒网页和当前访问网页的签名并计算签名的相似性，当相似性大于设定的阈值，则报警，提醒用户可能访问了钓鱼网页. 另外一种应用场景如下：浏览器插件从服务器下载特征库，该特征库中保存的是一些真实的电子商务网页的签名特征，它由服务器端维护更新；当用户访问一个需要输入密码的网页时将启动钓鱼网页检测功能，该功能将把当前访问网页的签名特征与特征库进行匹配，当发现钓鱼网页时发出报警.

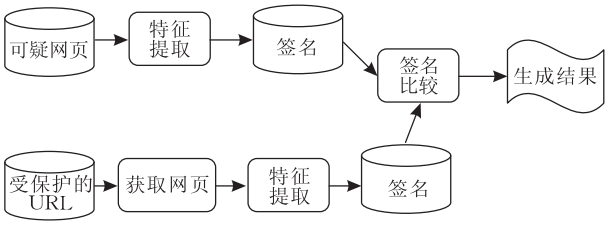



图 1 钓鱼网页检测过程

## 4 实 验

实验首先从钓鱼网页检测的精度、召回率两个方面作为基本评价指标，其次采用曲线下面积(*AUC*)指标来评估不同的特征、不同的匹配方法对检测结果的影响[19]. 记：

*A*：钓鱼网页被检测为钓鱼网页的数量；

*B*：正常网页被检测为钓鱼网页的数量；

*C*：钓鱼网页被检测为正常网页的数量；

*D*：正常网页被检测为正常网页的数量；

检测精度：$p=A/(A+B)$；

召回率：$r=A/(A+C)$；

正确肯定率(*TPR*)：$TPR=A/(A+C)$；

错误肯定率(*FPR*)：$FPR=B/(B+D)$.

精度描述了钓鱼网页检测的准确率，召回率描述了总的钓鱼网页中被检测出的比例. 这两个评价指标互相排斥，较高的精度意味着较低的召回率，而较低的精度意味着较高的召回率. 对于钓鱼网页检测来讲，更主要追求召回率指标，可以容忍一定比例的误报，毕竟提高安全性才是最主要的目的. 在实验中，我们把钓鱼网页作为正例，把正常网页作为反例. *TPR* 表示把真实的钓鱼网页预测为钓鱼网页的概率，*FPR* 表示把真实的正常网页错误地预测为钓鱼网页的概率. 设置不同的阈值来获得对应的 *TPR* 和 *FPR*，以此获得对应的 ROC 曲线，据此计算对应的 AUC 值.

本实验将做如下工作：(1) 确定文本特征(Text)相似性、图片(Image)特征相似性和网页 Overall 特征相似性计算过程中的参数；(2) 对这 3 种特种进行相关性分析；(3) 根据计算签名特征相似性时所采用方法的不同，本实验将比较 3 种不同的网页签名相似度计算方法对结果的影响：Simple、Normal 和 KM. (4) 对于这 3 种度量权重将采用 Logistic 回归[20]和 Probit 回归[21-22]分析方法来确定.

下面通过实例来说明这 3 种不同的相似度计算方法，设待计算相似性的签名特征 $S_1$ 和 $S_2$ 的长度分别为 $N_1$ 和 $N_2$. 例如，假设这两个签名特征的相似矩阵如下：

$$\boldsymbol{SM}=\begin{Bmatrix}0.1 & 0.3\\ 0.6 & 0.2\\ 0.9 & 0.8\end{Bmatrix},$$

其中 $N_1$ 为 3，$N_2$ 为 2.

Simple 方法. 根据相似矩阵求签名相似性时把相似矩阵的元素相加后取平均值，对于相似矩阵 ***SM***，算得签名 $S_1$ 和 $S_2$ 的相似性为 $(0.1+0.3+0.6+0.2+0.9+0.8)/6=0.48$. 其算法复杂度为 $O(N_1\times N_2)$.

Normal 方法. 首先选取矩阵中最大的元素(对 ***SM*** 来说为 0.9)，然后把该元素所在的行和列删除，重复这个步骤直到剩下的矩阵中为空，对于选取的元素相加后取平均值. 对于矩阵 ***SM*** 而言取到两个元素 0.9 和 0.3，则签名 $S_1$ 和 $S_2$ 的相似性为 $(0.9+0.3)/2=0.6$. 其算法复杂度为 $O(\min(N_1,N_2)\times N_1\times N_2)$.

KM 方法. 利用算法 1 通过求二分图的最大权和匹配，然后将匹配对的相似度取平均值. 对于 ***SM***，用算法 1 得到匹配的边的权重为 0.6 和 0.8，则签名 $S_1$ 和 $S_2$ 的相似性为 $(0.6+0.8)/2=0.7$. 其算法复杂度为 $O(\max(N_1,N_2)^3)$.

下文是具体的实验过程.

### 4.1 实验准备

由于大多数钓鱼网页的存在时间不长，所以没有直接从网上采集钓鱼网页. 另外也缺乏统一的、标准的钓鱼网页检测评价数据集，因此我们从网站(http://www.phishtank.com/phish_archive.php)采集样本，该网站是一个免费的反钓鱼网站，Web 用户提交可疑的钓鱼网页. 我们采集其中的部分网页，通过手工检查，去除合法网页以及其它一些噪音网页，总共收集 100 对钓鱼网页和对应的合法网页的 URL，把钓鱼网页内容和合法网页内容保存下来，另外根据 Yahoo 分类目录 bank、credit union、online services 等采集 100 个一般网页. 对于这些网页利用编写的 firefox 插件来遍历网页中的文本和图片以及获得网页整体图片，在此基础上提取网页签名(包括文本特征、图片特征和网页整体图片特征)，保存在特征表中. 实验中代码分别采用 javascript 和 visual studio 2008 C# 编写，运行于酷睿 2 笔记本电脑上，软硬件配置为 2.4GHz 酷睿处理器；2GB 内存；Windows XP Sp3，Mozilla FireFox 3.0 浏览器.

### 4.2　实验配置

在实验中，将 100 个被钓鱼的网页作为比较对象(特征库)，与之对应的 100 个钓鱼网页(正例)与其它 100 个一般网页(反例)构成样本库(容量为 200)．检测钓鱼网页的基本过程如下：将每个样本与特征库中的每个网页进行相似性计算，取相似度的最大值作为样本与特征库之间的相似度进行比较；如果相似度大于设定的阈值，则该网页检测为钓鱼网页，否则为一般性网页．这样意味着每个样本要与特征库中每个特征网页进行相似度计算，这样样本库遍历比较一次总共需要做 100×200 次比较，这无疑将是个很大的开销．为了提高匹配的速度，同时又不降低精度，在实验过程中利用一些特征进行快速过滤，比如网页标题相似性、网页文件大小、网页中的图片个数、网页 Overall 图像的面积等，记为

$Sim\_Title(P_1,P_2)$：网页$P_1$与$P_2$的标题相似性；

$Sim\_Size(P_1,P_2)$：网页 $P_1$与 $P_2$的文件大小相似性；

$Sim\_Images(P_1,P_2)$：网页 $P_1$与 $P_2$的图片个数相似性；

$Sim\_OverallArea(P_1,P_2)$：网页 $P_1$与 $P_2$的整体网页图片面积的相似性；

对于样本 $P$，遍历特征库中的每一个特征网页 $PL$，获得候选的特征样本集合 $SelectedLib$ 如下：

$SelectedLib$=NULL；

If(($Sim\_Title(P,PL)>Threshold$)

‖($Sim\_Size(P,PL)>Threshold$)

‖($Sim\_Images(P,PL)>Threshold$)

‖($Sim\_OverallArea\ (P,PL)>Threshold$))

$SelectedLib+=PL$；

其中，$Threshold$ 为设定的阈值，在实验中根据过滤效果我们设定其为 0.8．通过快速过滤，对于一个样本基本上可以控制特征样本集合 $SelectedLib$ 的大小在 10 个以下，从而大大提高了整体实验的速度．

在实验中为了比较特征权重对检测结果的影响，通过人工方式来设定一些参数的权重组合．另外为了从统计角度比较 Text 特征、Image 特征和 Overall 特征之间的关系以及比较 Simple 方法、Normal 方法和 KM 方法的差异，进行如下实验设计：对于文本特征的 6 个参数按照平均分布生成 210 组参数、对于图像特征的 5 个参数按照平均分布生成 125 组参数、而对于 Overll 特征的 3 个参数按照平均分布生成 10 组参数；对于文本特征、图像特征和 Overall 特征，分别采用 3 种匹配来检测钓鱼网页并计算对应参数下的 $AUC$ 值．根据 $AUC$ 指标来为 Text 特征、Image 特征和 Overall 特征选择最优的权重组合．另外根据不同参数组合下计算得到的 $AUC$ 值利用统计方法对不同匹配方法和不同特征进行相关性分析．在确定 3 种特征的内部权重后，利用 Logistic 回归和 Probit 回归来建立预测模型．

### 4.3　实验结果

为了比较 KM、Normal 和 Simple 这 3 种方法在性能上的差异以及在实际系统中的可行性，我们将这些匹配方法分别实现在 firefox 的插件中，统计在不同节点数情况下匹配所花费的时间，实验结果如图 2 所示．

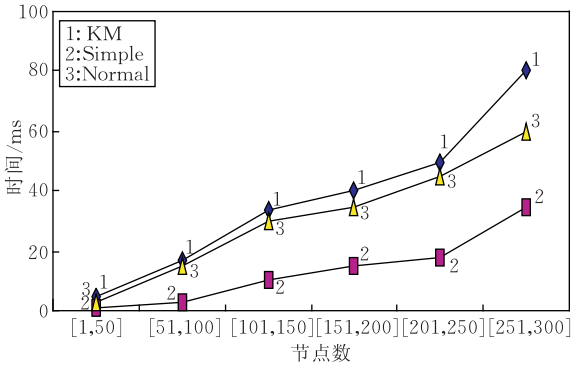



图 2　节点匹配的时间复杂度与节点数关系

实验中对节点数最多只统计到 300 个节点，这主要是对于大多数的网页来讲节点数在此之下；图 2 的结果表明，KM 方法的处理时间要大于 Normal 方法和 Simple 方法，Normal 方法要大于 Simple 方法，这些数据符合上文所分析的时间复杂性；通过对节点数在[251，300]之间的响应时间的分析，基于 KM 的匹配方法虽然时间复杂度较高，但在真实环境中在可容忍的范围之内．

#### 4.3.1　匹配方法及特征的比较

通过挑选 Simple、Normal、KM 方法中文本(Text)、图像(Image)和 Overall 对应 $AUC$ 的最大值(表 1(a))，可以看出：在 Simple 和 Normal 方法中，Overall 特征的效果最好，Image 特征次之，Text 特征最差；而在 KM 方法中，效果好坏的次序依次是：Overall 特征、Text 特征和 Image 特征．

从表 1(a)还可以看出：对于 Text 特征和 Image 特征，$AUC$ 值从大到小的方法依次是：KM 方法、Normal 方法和 Simple 方法；而对于 Overall 特征，由于未采用匹配方法，所以具有相同的 $AUC$ 值．表 1(a)中的最大的 $AUC$ 值为对应于 Text 特征采用 KM 方法．表 1(b)为不同匹配方法和不同特征所对应最大的 $AUC$ 值下所对应的权重．

**表 1 不同方法 *AUC* 值及权重确定**

(a) Simple、Normal、KM 方法中 Text、Image 和 Overall 中所对应的最大 *AUC* 值

| | Text | Image | Overall |
|---|---|---|---|
| Simple | 0.8481 | 0.8881 | 0.9773 |
| Normal | 0.9054 | 0.9231 | 0.9773 |
| KM | 0.98758 | 0.9597 | 0.9773 |

(b) Simple、Normal、KM 方法中根据最大 *AUC* 值所得到的 Text、Image 和 Overall 特征的权重

| | Text | | | Image | | Overall |
|---|---|---|---|---|---|---|
| Simple | | 0.11 | 0.23 | | 0.15 | 0.50 |
| | 0.23 | 0.11 | 0.23 | 0.28 | 0.15 | 0.50 |
| | | 0.11 | | 0.15 | 0.27 | |
| Normal | | 0.09 | 0.57 | | 0.10 | 0.50 |
| | 0.09 | 0.09 | 0.09 | 0.48 | 0.22 | 0.50 |
| | | 0.09 | | 0.10 | 0.10 | |
| KM | | 0.09 | 0.21 | | 0.10 | 0.50 |
| | 0.09 | 0.09 | 0.08 | 0.48 | 0.10 | 0.50 |
| | | 0.44 | | 0.10 | 0.22 | |

表 2(a)和表 2(b)分别针对 Text 特征和 Image 特征的 3 种匹配方法(Simple、Normal 和 KM)的 *AUC* 值的描述统计. 从平均值可以看出,KM 方法具有最好的效果. Simple 方法效果最差,这主要是由于 Simple 方法中的匹配方法将相似节点的相似度与不相似节点的相似度平均化,降低了对相似网页与不相似网页的区分能力. 另外,我们可以看出 Normal 方法比 Simpe 方法的效果要好,这主要是由于 Normal 方法从相似矩阵中取最大的几个相似对,一定程度上提高了计算的精度. KM 方法在匹配过程中通过求最佳匹配获得用于计算相似性的相似对,这些相似对基本上包含了 Normal 方法中的相似对.

**表 2 Text 与 Image 特征 *AUC* 描述统计**

(a) Text 特征

| | 平均 | 标准误差 | 中位数 | 标准差 | 方差 | 最小值 | 最大值 | 观测数 | 置信度(95.0%) |
|---|---|---|---|---|---|---|---|---|---|
| Simple | 0.8069 | 0.0029 | 0.8206 | 0.0416 | 0.0017 | 0.5758 | 0.8481 | 210 | 0.0057 |
| Normal | 0.8665 | 0.0009 | 0.8657 | 0.0126 | 0.0001 | 0.8364 | 0.9054 | 210 | 0.0017 |
| KM | 0.9679 | 0.0005 | 0.9662 | 0.0075 | 0 | 0.9534 | 0.9876 | 210 | 0.001 |

(b) Image 特征

| | 平均 | 标准误差 | 中位数 | 标准差 | 方差 | 最小值 | 最大值 | 观测数 | 置信度(95.0%) |
|---|---|---|---|---|---|---|---|---|---|
| Simple | 0.8257 | 0.002 | 0.8268 | 0.0223 | 0.0005 | 0.6718 | 0.8881 | 125 | 0.0039 |
| Normal | 0.8905 | 0.0012 | 0.8901 | 0.0137 | 0.0002 | 0.863 | 0.9231 | 125 | 0.0024 |
| KM | 0.9462 | 0.0005 | 0.9456 | 0.0058 | 0 | 0.9311 | 0.9597 | 125 | 0.001 |

下面从统计角度分析 3 种匹配方法的差异. 对于文本特征,Simple、Normal 和 KM 方法的差别比较可以通过对 3 种方法的 *AUC* 值进行两两的成对双样本均值分析 $t$ 检验,检验结果如表 3(a)所示,其中上三角矩阵的值为 $t$ 值,下三角矩阵的值为对应的 $p$ 值. 从该表中可以看出,Simple、Normal 和 KM 方法两两之间具有显著差异. 这主要由于 KM 方法中的节点匹配后的相似度度量要优于 Normal 方法中取前几个最大相似节点的相似度度量,这两种方法又要优于 Simple 方法中相似度取平均的度量. 表 3(b)给出了对于图像特征 3 种方法的 *AUC* 值显著性 $t$ 检验,与表 3(a)具有同样的结果,充分说明了基于 KM 方法的相似度度量的优越性. 对于 Overall 特征,由于一个网页对应一个 Overall 特征,所以不存在匹配问题,也就无法根据该特征来考察 3 种匹配方法的优劣. 表 4(a)和(b)中显示利用非参数假设检验——威尔科克逊秩和检验的结果,得到了与 $t$ 检验同样的结果,即 Simple、Normal 和 KM 方法存在显著差异.

**表 3 *AUC* 值显著性 $t$ 检验**

(a) 对于文本特征 3 种方法的 *AUC* 值显著性 $t$ 检验

| | Simple | Normal | KM |
|---|---|---|---|
| Simple | | −19.223 | −25.643 |
| Normal | <0.001 | | −7.514 |
| KM | <0.001 | <0.001 | |

(b) 对于图像特征 3 种方法的 AUC 值显著性 $t$ 检验

| | Simple | Normal | KM |
|---|---|---|---|
| Simple | | −8.274 | −9.677 |
| Normal | <0.001 | | −9.338 |
| KM | <0.001 | <0.001 | |

**表 4 威尔科克逊秩和检验**

(a) 对于文本特征 3 种方法的威尔科克逊秩和检验

| | Simple | Normal | KM |
|---|---|---|---|
| Simple | | 1830 | 1893 |
| Normal | 0 | | 2757.5 |
| KM | 0 | <0.001 | |

(b) 对于图像特征 3 种方法的威尔科克逊秩和检验

| | Simple | Normal | KM |
|---|---|---|---|
| Simple | | 1700 | 1525 |
| Normal | <0.001 | | 2101.5 |
| KM | <0.001 | 0.0035 | |

表 5(a)～(c)分别表示对于 3 种不同的匹配方法，3 种特征之间的差异性. 对于 Simple 匹配方法，3 种特征之间有显著差异；对于 Normal 和 KM 匹配方法，Text 特征与 Image 特征和 Overall 特征之间存在显著差异，而 Image 特征与 Overall 特征之间不存在显著差异.

**表 5　不同匹配方法的 $t$ 检验**

(a) 对于 Simple，$t$ 检验

| | Text | Image | Overall |
|---|---|---|---|
| Text | | 5.2656 | −9.2221 |
| Image | <0.001 | | −15.2202 |
| Overall | <0.001 | <0.001 | |

(b) 对于 Normal，$t$ 检验

| | Text | Image | Overall |
|---|---|---|---|
| Text | | 18.5628 | 30.8529 |
| Image | <0.001 | | −2.4709 |
| Overall | <0.001 | 0.01634 | |

(c) 对于 KM，$t$ 检验

| | Text | Image | Overall |
|---|---|---|---|
| Text | | 14.9697 | 33.9305 |
| Image | <0.001 | | 1.4829 |
| Overall | <0.001 | 0.1437 | |

表 5(a)～(c)均是假设总体服从正态分布的情况下获得的结果. 为了放宽这种假设条件，引入非参数假设检验——威尔科克逊秩和检验，检验结果如表 6(a)～(c)(上三角为 $t1$ 统计量：数据个数较少的样本的秩和，下三角为 $p$ 值)所示，对于 Simple 和 KM 方法，Text 特征、Image 特征和 Overall 特征之间存在显著差异；而对于 Normal 方法，Text 特征与 Image 特征和 Overall 特征存在显著差异，但是 Image 特征和 Overall 特征不存在显著差异.

**表 6　不同匹配方法的秩和检验**

(a) 对于 Simple 威尔科克逊秩和检验

| | Text | Image | Overall |
|---|---|---|---|
| Text | | 1989 | 1312.5 |
| Image | <0.001 | | 1210 |
| Overall | <0.001 | <0.001 | |

(b) 对于 Normal 威尔科克逊秩和检验

| | Text | Image | Overall |
|---|---|---|---|
| Text | | 1341.5 | 210 |
| Image | 0 | | 922.5 |
| Overall | <0.001 | 0.0057 | |

(c) 对于 KM 威尔科克逊秩和检验

| | Text | Image | Overall |
|---|---|---|---|
| Text | | 1371 | 210 |
| Image | 0 | | 684 |
| Overall | <0.001 | <0.001 | |

4.3.2　预测模型

Text 特征、Image 特征和 Overall 特征均可用于钓鱼网页检测，上文的实验已经获得每种特征内部的最优权重. 在实际检测中，需要综合使用这 3 种特征，这就需要确定这 3 种特征的相对重要程度. 合理的相对权重对于提高综合评价钓鱼网页的精度非常关键. 这里先对 KM 方法分别采用 Logistic (Logit)和 Probit 回归模型建模，并采用 AIC 方法[23-24]和 BIC 方法[25]进行模型选择，分别记为 KM Logit ALL、KM Logit AIC、KM Logit BIC、KM Probit ALL、KM Probit AIC 和 KM Probit BIC.

(1) 描述性分析

考虑到因变量与自变量的关系，我们利用盒状图考察因变量(是否为钓鱼网页)与自变量(由 Text 特征、Image 特征和 Overall 特征所计算得到的与特征库的相似度)的关系. 从图 3 中我们可以得到如下初步结论：

① 钓鱼网页文本特征的平均相似度要明显高于正常网页文本特征的相似度.

② 钓鱼网页图像特征的平均相似度要明显大于正常网页的图像特征的相似度.

③ 钓鱼网页 Overall 特征的平均相似度要大于正常网页的 Overall 特征的相似度，钓鱼网页的 Overall 特征相似度落在一个很小的区间内，而正常网页的 Overall 特征相似度的平均值也大于 0.9，另外在该图中还存在一些异常点. 导致正常网页的 Overall 特征的相似度较大以及几个异常点的存在跟所提取的 Overall 中具体的图像特征有关.

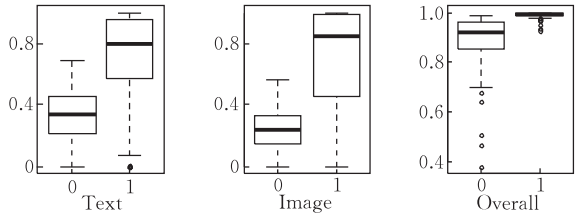



图 3　盒状图

(2) 数据建模

① 全模型分析

根据因变量为 0-1 变量的特征，我们采用 Probit 回归和 Logistic 回归的方法建立模型. 首先对包含全部自变量的全模型进行 Logistic 估计，得到的结果如表 7(a)所示.

**表 7 KM 模型选择**

(a) KM Logit ALL

| 变量名 | 系数估计值 | 标准差 | $p$ 值 |
|---|---|---|---|
| 截距 | −110.366 | 24.134 | <0.001 |
| Text | 2.496 | 1.337 | 0.061972 |
| Image | 6.551 | 1.866 | <0.001 |
| Overall | 110.229 | 24.546 | <0.001 |
| 模型 $F$ 检验 $p$ 值<0.001 | | | |

(b) KM Probit ALL

| 变量名 | 系数估计值 | 标准差 | $p$ 值 |
|---|---|---|---|
| 截距 | −57.5977 | 11.6252 | <0.001 |
| Text | 1.2696 | 0.7002 | 0.069805 |
| Image | 3.6901 | 0.9761 | <0.001 |
| Overall | 57.5259 | 11.8663 | <0.001 |
| 模型 $F$ 检验 $p$ 值<0.001 | | | |

从表 7(a)中我们可以看到,对模型 $F$ 检验的 $p$ 值非常小($p$ 值<0.001),表明该模型是显著的,即我们所考虑的 3 个解释性变量中,至少有一个与是否为钓鱼网页显著相关.进一步对各自变量所对应的 $Z$ 检验的考察,我们可以得到以下重要结论.

Image 特征与 Overall 特征和因变量高度正相关,而 Text 特征与因变量相关性相对于另外两个特征较弱.这与实验之前我们的主观判断有一定的差异.这说明,钓鱼网页在制作的时候主要是满足视觉效果上的相似性以达到欺骗用户的目的,而在文本方面可以有较大的出入(有些钓鱼网页直接复制被钓鱼网页的整体图片,而文本较少).

3 个解释性变量(Text、Image 和 Overall)的影响力很不一样.Overall 特征的系数非常大,Image 特征的系数其次,Text 特征的系数最小,这在一定程度上说明,Overall 特征(反映整体视觉效果)对于检测钓鱼网页的重要性.这也为钓鱼网页的检测指明了一个方向:应该更加注重视觉相似度的检测.

类似地,我们用 Probit 回归分析,得到回归结果如表 7(b)所示.

从表 7(b)可以发现,基本结论同 KM Logit ALL 模型.

(2) 模型选择

上文的分析结果表明,我们所选取的自变量确实对因变量有一定的解释能力.为了得到一个尽量简单同时又有良好预测能力的模型,我们采用 AIC 和 BIC 的模型选择标准来选择一个最优的模型.对于 Logistic 回归模型,我们用 AIC 方法和 BIC 方法选出模型及其估计结果分别如表 8(a)和表 8(b)所示,表 8(a)结果与表 8(b)全模型 Logistic 回归相同.从表 8(a)及表 8(b)中我们可以发现,AIC 与 BIC 得出了不一样的结论.AIC 认为 3 个因素(Text 特征、Image 特征和 Overall 特征)与是否为钓鱼网页相关,而 BIC 认为只有 Image 特征和 Overall 特征是重要的.从一个保守的角度,我们可以得到如下结论:3 个因素都很重要,而其中 Image 特征和 Overall 特征格外重要.通过对 Probit 回归方法作类似的模型选择,我们发现两种回归方法的模型选择结果完全一致.因此,我们认为上述的模型选择结果是可靠的.对于本文所提供的实验数据,AIC 和 BIC 得到了相同的结论,它们都认为全模型就是最优模型.这表明,这 3 个自变量确实都有一定的预测能力.

**表 8 KM 模型选择**

(a) KM Logit BIC

| 变量名 | 系数估计值 | 标准差 | $p$ 值 |
|---|---|---|---|
| 截距 | −110.366 | 24.134 | <0.001 |
| Text | 2.496 | 1.337 | 0.061972 |
| Image | 6.551 | 1.866 | <0.001 |
| Overall | 110.229 | 24.546 | <0.001 |
| 模型 $F$ 检验 $p$ 值<0.001 | | | |

(b) KM Logit BIC

| 变量名 | 系数估计值 | 标准差 | $p$ 值 |
|---|---|---|---|
| 截距 | −118.383 | 24.402 | <0.001 |
| Image | 7.519 | 1.874 | <0.001 |
| Overall | 119.331 | 24.729 | <0.001 |
| 模型 $F$ 检验 $p$ 值<0.001 | | | |

(3) 模型预测与评估

通过预测的结果,对刚才分析得到的 KM 方法的六种预测模型进行比较.这 6 个模型分别为 KM Logistic 回归全模型(KM Logit ALL)、KM Logistic 回归的 AIC 模型(KM Logit AIC)、KM Logistic 回归的 BIC 模型(KM Logit BIC)、KM Probit 回归全模型(KM Probit ALL)、KM Probit 回归的 AIC 模型(KM Probit AIC)、KM Probit 回归的 BIC 模型(KM Probit BIC).

通过对原有数据的预测,并用对应的预测结果来衡量模型的预测精度.我们采用 $TPR$ 和 $FPR$ 两个指标间接地度量预测精度.

对于 Logistic 回归分析,需要先给定一个阈值才能进行预测.而不同的阈值所对应的预测结果和精度也不同.另外,由于高的 $TPR$ 值总是对应着高的 $FPR$ 值,所以只能根据风险偏好来选择合适的 $TPR$ 和 $FPR$ 值.因此,这里采用 ROC 曲线来度量模型之间的差异,如图 4 所示.

图 4 中的 ROC 曲线与对角线比,是上凸的.这

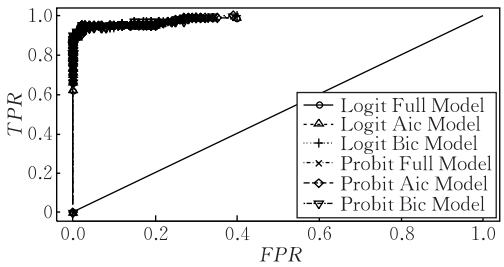



图 4　KM 六种模型的 ROC 曲线图

表明了预测模型是有效的. 从图中可以明显地看到，6 个模型的的 ROC 曲线非常相似，这说明它们的预测能力相当.

从上述的分析结果可知，文本特征、图像特征和 Overall 特征都影响着钓鱼网页的检测，其中 Overall 特征的影响要大于图像特征和文本特征，而图像特征又要大于文本特征. 因此对于钓鱼网页检测来说，可以利用 Overall 特征首先进行快速过滤，在确定疑似钓鱼网页以后再结合文本特征和图像特征来进一步检测.

对于 Simple 匹配方法和 Normal 匹配方法类似得到图 5 和图 6 所示的 ROC 曲线图，从图可知对于 Simple 方法和 Normal 方法，分别的 6 种模型的预测能力相当.

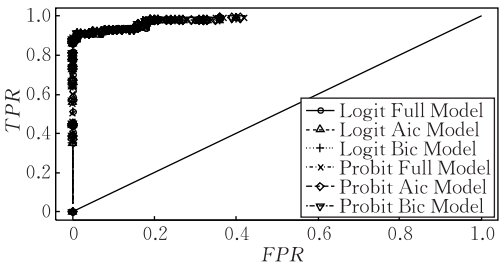



图 5　Simple 六种模型的 ROC 曲线图

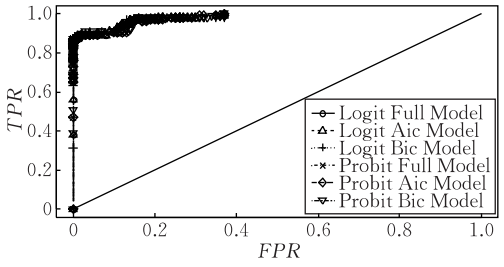



图 6　Normal 六种模型的 ROC 曲线图

为了比较文中所提各种模型在具体效果上的差异，表 9 给出了每种模型在其最优参数下的精度、召回率和 $F1$-Measure. 从表中可以看出 KM Logit ALL、KM Logit AIC、KM Probit ALL 和 KM Probit AIC 具有最好的 $F1$-Measure. 另外 KM 方法的组合模型在 $F1$-Measure 上都要优于单独采用某种特征的模型. 对于 KM 和 Simple 方法，在 3 种单独模型中，Overall 效果最好，而 Normal 方法的 3 种单独模型中，text 特征效果最好，这说明匹配方法对于特征的效果有影响.

**表 9　不同模型在最优阈值下的模型、召回率和 $F1$-Measure**

| 模型 | 阈值 | 精度 | 召回率 | $F1$-Measure |
|---|---|---|---|---|
| KM Logit ALL&AIC | 0.491 | 0.979 | 0.950 | 0.964 * |
| KM Logit BIC | 0.702 | 1.000 | 0.920 | 0.958 |
| KM Probit ALL&AIC | 0.503 | 0.979 | 0.950 | 0.964 * |
| KM Probit BIC | 0.633 | 0.979 | 0.930 | 0.954 |
| KM Text | 0.560 | 0.950 | 0.784 | 0.859 |
| KM Image | 0.408 | 0.909 | 0.860 | 0.884 |
| KM Overall | 0.980 | 0.978 | 0.890 | 0.932 |
| Simple Logit ALL | 0.660 | 0.989 | 0.910 | 0.948 |
| Simple Logic AIC&BIC | 0.687 | 0.989 | 0.910 | 0.948 |
| Simple Probit ALL | 0.670 | 0.989 | 0.910 | 0.948 |
| Simple Probit AIC&BIC | 0.694 | 0.989 | 0.910 | 0.948 |
| Simple Text | 0.457 | 0.809 | 0.800 | 0.804 |
| Simple Image | 0.159 | 0.803 | 0.980 | 0.883 |
| Simple Overall | 0.980 | 0.978 | 0.890 | 0.932 |
| Normal Logit ALL&AIC | 0.647 | 0.978 | 0.890 | 0.932 |
| Normal Logit BIC | 0.605 | 0.968 | 0.910 | 0.938 |
| Normal Probit ALL&AIC | 0.670 | 0.989 | 0.890 | 0.937 |
| Normal Probit BIC | 0.621 | 0.968 | 0.910 | 0.938 |
| Normal Text | 0.970 | 1.000 | 0.898 | 0.946 |
| Normal Image | 0.773 | 0.977 | 0.885 | 0.929 |
| Normal Overall | 0.980 | 0.978 | 0.890 | 0.932 |

### 4.4　讨　论

从表 1(a)、(b)和表 2(a)、(b)可以看出，KM 匹配算法相对于 Simple 方法和 Normal 方法具有更好的效果. 这主要是由于 KM 方法通过寻找节点之间的最佳匹配来作为进一步计算相似度的依据，避免了 Simple 方法的随意性；而 Normal 方法通过取相似矩阵的最大行列值的做法实际上是把最相似的几个值的均值作为相似值，这无疑忽略了其它节点的作用，容易导致相似度偏大.

表 7 和表 8 的结果表明，Overall 特征的效果最好，Image 特征的效果其次，而 Text 特征的效果最差，这主要是跟钓鱼网页的构造过程有关，它主要在视觉上保证相似性，以此来混淆用户的判断. 对于视觉效果相同的网页(Overall 特征)，存在着相似度计算偏低的问题. 这主要是由于很多钓鱼网页直接通过截取被模仿网页的图片，然后在此基础上制作网页. 这就导致钓鱼网页与被模仿网页在文本特征和图片特征上几乎没有相似性，而只是在 Overall 上存在相似性. 表 9 表明 KM 方法的全模型具有最优的效果.

## 5 结束语

在确保钓鱼网页检测召回率的情况下,提高钓鱼网页的检测精度是提高钓鱼网页检测效果的目标.本文从 Text 特征相似性、Image 特征相似性和网页 Overall 相似性 3 个角度来刻画网页之间的相似性,并通过匹配方法实现不等长特征相似性计算.本文的主要贡献包括:(1)提出了基于匈牙利匹配的网页签名相似度计算方法,在匹配特征对基础上计算出的相似性更客观.(2)给出了具体的网页签名提取方法,包括网页 Text 签名特征、Image 签名特征和网页 Overall 特征,通过实验确定了各种特征之间的相对权重,并从统计角度分析了文本特征、图像特征和 Overall 特征之间的差异以及文中 3 种匹配方法之间的差异.(3)给出了组合多种特征的最优回归模型 KM logit ALL 和 KM probit ALL.本文将来的工作主要集中于如何提高网页图片匹配的效果(选择合适的图像特征)、如何构建钓鱼网页特征库索引以进一步提高检测速度、如何实现设置 3 种特征的动态权重等.

**致 谢** 在此,我们向对本文的工作给予支持和建议的同行表示感谢!

## 参 考 文 献


[1] Abu-Nimeh S, Nappa D, Wang X, Nair S. A comparison of machine learning techniques for phishing detection//Proceedings of the eCrime Researchers Summit. Pittsburgh, PA, USA, 2007: 60-69

[2] Zhang Y, Hong J, Cranor L F. Cantina: A content-based approach to detecting phishing web sites//Proceedings of the International Conference on World Wide Web. Banff, Alberta, Canada, 2007: 639-648

[3] Kumaraguru P, Sheng S, Acquisti A, Cranor L F, Hong J. Teaching Johnny not to fall for phish. ACM Transactions on Internet Technology, 2010, 10(2): 1-31

[4] Sheng S, Holbrook M, Kumaraguru P, Cranor L F, Downs J. Who falls for phish?: A demographic analysis of phishing susceptibility and effectiveness of interventions//Proceedings of the 28th International Conference on Human Factors in Computing Systems. Atlanta, Georgia, USA, 2010: 373-382

[5] Schneider F, Provos N, Moll, R, Chew, M, and Rakowski B. Phishing protection design documentation. http://wiki.mozilla.org/Phishing_Protection:_Design_Documentation, 2007

[6] NetCraft. Netcraft Anti-Phishing tool bar. http:// toolbar.netcraft.com, 2007

[7] McAfee. McAfee SiteAdvisor. http: //www. siteadvisor.com, 2007

[8] Dhamija R, Tygar J D. The battle against phishing: Dynamic securityskins//Proceedings of the Symposium on Usable Privacy and Security. Pittsburgh, Pennsylvania, 2005: 77-88

[9] Liu W, Huang G, Liu X, Zhang M, Deng X. Detection of phishing Web pages based on visual similarity//Proceedings of 14th International World Wide Web Conference. Chiba, Japan, 2005: 1060-1061

[10] Liu W, Deng X, Huang G, Fu A Y. An anti-Phishing strategy based on visual similarity assessment. IEEE Internet Computing, 2006, 10(2): 58-65

[11] Kumaraguru P, Sheng S, Acquisti A, Cranor L F, Hong J. Teaching Johnny not to fall for phish. ACM Transactions on Internet Technology, 2010, 10(2): 1-31

[12] Dunlop M, Groat S, Shelly D. GoldPhish: Using images for content-based phishing analysis//Proceedings of the 5th International Conference on Internet Monitoring and Protection. Barcelona, Spain, 2010: 123-128

[13] Rosiello A, Kirda E, Kruegel C, Ferrandi F. A layout-similarity-based approach for detecting phishing pages//Proceedings of the International Conference on Security and Privacy in Communication Networks. Nice, France, 2007: 454-463

[14] Medvet E, Kirda E, Kruegel C. Visual-similarity-based phishing detecting//Proceedings of the 4th International Conference on Security and Privacy in Communication Networks. Istanbul, Turkey, 2008: 1-6

[15] Fu A Y, Liu W, Deng X. Detecting phishing web pages with visual similarity assessment based on earth mover's distance (EMD). IEEE Transactions on Dependable and Secure Computing, 2006, 3(4): 301-311

[16] Cao Jiu-Xin, Mao Bo, Luo Jun-Zhou, Liu Bo. A phishing web pages detection algorithm based on nested structure of each mover's distance (nested-EMD). Chinese Journal of Computers, 2009, 32(5): 922-929
(曹玖新, 毛波, 罗军舟, 刘波. 基于嵌套 EMD 的钓鱼网页检测算法. 计算机学报, 2009, 32(5): 922-929)

[17] Lovasz L, Plummer M. Matching Theory. Amsterdam: North-Holland, 1986

[18] Kuhn H. The Hungarian method for the assignment problem. Naval Research Logistics, 2005, 2(1): 7-21

[19] Jiang Y, Cukic B, Ma Y. Techniques for evaluating fault prediction models. Empirical Software Engineering, 2008, 13(5): 561-595

[20] Hilbe Joseph M. Logistic Regression Models. Boca Raton, FL:Chapman & Hall/CRC Press, 2009

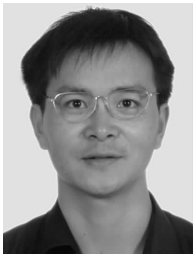
[21] Bliss C I. The determination of the dosage-mortality curve from small numbers. Quarterly Journal of Pharmacology, 1938, 11: 192-216

[22] McCullagh P, Nelder J. Generalized Linear Models. London: Chapman and Hall, 1989

[23] Hirotugu Akaike. A new look at the statistical model identification. IEEE Transactions on Automatic Control, 2009, 19 (6): 716-723

[24] Burnham K P, Anderson D R. Model Selection and Multimodel Inference: A Practical Information-Theoretic Approach. 2nd Edition. New York: Springer-Verlag, 2002

[25] McQuarrie A D R, Tsai C-L. Regression and Time Series Model Selection. Singapore: World Scientific Publishing Company, 1998



**ZHANG Wei-Feng**, born in 1975, Ph. D., associate professor. His research interests include web information retrieval, web data mining, and spam detecting technology.

**ZHOU Yu-Ming**, born in 1974, Ph.D., professor. His research interests include software metrics and software testing.

**XU Lei**, born in 1974, Ph. D., associate professor. Her research interests include web testing, web service, and web technology.

**XU Bao-Wen**, born in 1961, professor. His research interests include software metrics, software testing, and web technology.

**Background**

Along with the popularization of Internet applications, more and more people have been accustomed to various online Internet services such as online banking, online shopping, etc. In the mean time, the number of phishing web sites aiming to steal the sensitive information of the victims has been rapidly increased. Phishing web pages are the fake web pages created intentionally by some criminals, who copy web pages from real web sites. Therefore, most phishing web pages have high visual similarities to their real counterparts. In general, phishing web sites deceive Internet users by mimicking the interface of their real counterparts. When a user enters in a phishing site, sensitive information, such as user name, password, bank account, credit card number or other important personal information, such information will be stolen and may be illegally used by the phishing web page owners. This is very likely to result in huge loss to the users.

Near duplication detection is an effective scheme in phishing detection. In contrast to previous methods each of which uses a simple similarity scheme on the features, image features, or overall features, this paper enhances the performance of phishing detection by using a novel approach that synthetically exploits all the three feature types. The approach uses Hungarian matching algorithm to calculate the similarity between web pages, and adapts regression methods to compute the optimal weights of the three feature types. Specifically, the approach first collects signatures of web pages, i. e. text features, image features, and overall features, from the pages rendered in the browser, by which the overall features from viewpoint can be characterized. Second, the Hungarian algorithm is used to compare the web page signatures, which enhances the effect of similarity comparison. Last, regression models are built up to seek the optimal weights for the three feature types and to seek an appropriate model to detect phishing. Experiments show that it is beneficial to exploit all the three feature types synthetically and to compute similarity using the Hungarian algorithm, as the proposed method achieves better performance than some existing methods.